PENDING CLAIMS

Although no amendments are being made to the claims by this Response, a copy of the claims pending in this application are given below for the convenience of the Examiner:

1. (Original) A method of initiating a memory storage system having flash memory containing at least first and second copies of firmware code stored in different locations therein, a microprocessor, a read-only-memory (ROM) containing microprocessor accessible boot code and a random-access-memory (RAM) for storing microprocessor accessible firmware code, the method comprising:

executing the boot code to transfer a first copy of the firmware from the flash memory to the RAM,

identifying any bit errors in the transferred first copy of the firmware code,

if bit errors are identified that are correctable, correcting the erroneous bits,

if bit errors are identified that are not correctable, reading at least a portion of the second copy of the firmware code into the RAM in place of at least a portion of the first copy containing the uncorrectable bit errors, and

executing an error free copy of the firmware code from the RAM.

2. (Original) The method of claim 1, wherein identifying any bit errors in the transferred first copy includes calculating error-correction-codes (ECCs) from individual portions of the first copy of the firmware by passing the firmware portions through ECC circuitry in succession as they are being transferred from the flash memory to the RAM, and comparing the calculated ECCs with ECCs previously calculated from said portions of the first copy of the firmware data.

3. (Original) The method of claim 2, wherein correcting the erroneous bits includes the microprocessor executing an error correction algorithm of the boot code to correct erroneous bits.

4. (Original) The method of claim 2, wherein the individual portions of the first copy of the firmware code include one or more sectors of data and an ECC previously calculated therefrom and stored in the flash memory therewith.

5. (Original) The method of claim 1, which additionally comprises, prior to executing the boot code to transfer a first copy of the firmware from the flash memory to the RAM, the following:

initially accessing a plurality of fixed locations in the flash memory one at a time until an initialization memory map is discovered to be stored at at least one of the plurality of fixed locations and that contains addresses of the different locations of the flash memory wherein said at least first and second copies of firmware code are stored,

reading data of the initialization memory map to obtain said addresses, and

thereafter accessing the first copy of the firmware code.

6. (Original) The method of claim 5, which additionally comprises:

identifying any bit errors in the data read from the initialization memory map,

if bit errors in the data read are identified that are correctable, correcting the erroneous bits, and

if bit errors in the data read are identified that are not correctable, re-reading the data of the initialization memory map under different conditions.

7. (Original) The method of claim 1, wherein the at least first and second copies of firmware code are stored in the different locations of the flash memory with only a given number of one or more bits of firmware code stored per flash memory storage element thereof, while the memory storage system is further characterized by storing user data in other locations of the flash memory with more than said given number of bits of user data per storage element thereof.

8. (Original) The method of claim 7, wherein said given number of one or more bits is exactly one bit per flash memory storage element.

9. (Original) The method of claim 1, additionally comprising:

identifying any bit errors in the transferred at least a portion of the second copy of the firmware code, and

if bit errors identified in the transferred at least a portion of the second copy of the firmware code are not correctable, repeating the reading of said at least a portion of the second copy of the firmware code under conditions that tend to reduce the number of bit errors in the transferred at least a portion of the second copy.

10. (Original) The method of claim 1, additionally comprising, prior to executing the boot code to transfer a first copy of the firmware from the flash memory to the RAM, checking the state of a firmware present flag that is set when firmware is stored in the flash memory and continuing to execute the boot code to transfer the first copy of the firmware from the flash memory to the RAM only when the firmware present flag is set.

11. (Original) The method of claim 1, additionally comprising, in response to identifying a number of bit errors of a predefined number of one or more, of setting a housekeeping flag associated with the locations of the flash memory from which the erroneous data of the first copy of the firmware are stored.

12. (Original) The method of claim 11, additionally comprising, in response to the housekeeping flag being set, of correcting the erroneous data of the first copy of the firmware after an error free copy of the firmware code has been transferred into the RAM.

13. (Original) The method of claim 12, wherein correcting the erroneous data of the first copy of the firmware includes rewriting the corrected first copy of the firmware in the flash memory.

14. (Original) The method of claim 13, wherein rewriting the corrected first copy of the firmware includes re-writing the corrected first copy in a different location than it was originally stored.

15. (Original) The method of claim 12, wherein correcting the erroneous data of the first copy of firmware includes use of an error-correction-code.

16. (Original) The method of claim 12, wherein correcting the erroneous data of the first copy of firmware includes transfer of good data from the second copy of firmware code.

17. (Original) A method of operating a memory storage system having flash memory, a microprocessor, a read-only-memory (ROM) containing boot code accessible by the microprocessor, a random-access-memory (RAM) and circuitry that calculates an error correction code (ECC) from data passing through it, the method comprising:

storing at least first and second copies of firmware code in different addressable locations of the flash memory by passing the firmware copies one at a time through the ECC circuitry and storing the ECCs calculated thereby in the flash memory,

thereafter initiating operation of the memory system by causing the microprocessor to execute the boot code to transfer the first copy of the firmware from the flash memory to the RAM through the ECC circuitry which calculates an ECC therefrom,

utilizing the calculated and stored ECCs to identify any bit errors in the transferred first copy of the firmware code, and

if bit errors are identified to be correctable, causing the microprocessor to execute an error correction algorithm within the boot code to correct the erroneous bits, in order to result in the firmware code being loaded into the RAM without any errors, or

if bit errors are identified to be uncorrectable, transferring at least a portion of the second copy of the firmware code into the RAM in place of at least a portion of the first copy containing the uncorrectable bit errors, in order to result in the firmware code being loaded into the RAM without any errors.

18. (Original) The method of claim 17, wherein storing the firmware code includes storing ECCs individually calculated from one or more sectors of the firmware code.

19. (Original) The method of claim 17, additionally comprising storing a map in one of a predetermined plurality of locations of the flash memory that contains the addressable locations of said at least first and second copies of firmware code, and wherein execution of the boot code by the microprocessor includes initially locating the map by accessing the predetermined plurality of locations one at a time until the map is found, and reading the contents of the map at the location where the map is stored.

20. (Original) The method of claims 17, wherein storing the firmware code additionally includes setting a flag to indicate the presence within the flash memory of at least one firmware copy, and wherein executing the boot code to transfer either of the first or second copies of the firmware code includes first reading the flag associated therewith and proceeding to read the copy of the firmware code only if the associated flag is set.

21. (Original) A flash memory storage system, comprising:

an array of flash memory cells storing data in charge storage elements and containing a memory map at at least one of a plurality of predetermined addresses of the array, said map including data specifying addresses wherein one or more copies of firmware code are stored,

a controller processor,

a read-only-memory containing boot code that the processor accesses and executes in response to initialization of the storage system,

a random-access-memory that is accessible by the processor to obtain instructions to be executed, and

wherein the boot code causes the processor to access the plurality of predetermined addresses within the flash memory to locate and read the data of the memory map specifying addresses wherein one or more copies of the firmware are stored, thereafter reading the firmware code located at at least one of said specified one or more addresses and thereafter writing the read the firmware code into the random-access-memory.

22. (Original) The system of claim 21, wherein the map and the firmware code are stored in the flash memory with one-bit thereof per memory cell storage element, and further wherein data are stored at at least some addresses of the memory array other than those containing the map and firmware with more than one-bit thereof per memory cell storage element.

23. (Original) A flash memory storage system, comprising:

an array of flash memory cells storing data in charge storage elements and containing at least first and second copies of firmware code stored therein along with respective first and second sets of error-correction codes (ECCs) calculated from the first and second copies of the firmware code,

a controller processor,

circuitry that calculates ECCs from data passing through the circuitry,

a read-only-memory containing boot code that the processor accesses and executes in response to initialization of the storage system,

a random-access-memory that is accessible by the processor to obtain instructions to be executed, and

wherein the boot code causes the processor to read the first firmware code copy including passing the read first firmware code copy through the ECC calculation circuitry which calculates ECCs and provides with respect to the first set of ECCs stored with the first firmware code copy a status with respect to any data errors existing in portions of the first firmware code copy to which the ECCs pertain, and

(A) if the status indicates that there are no data errors in a given one of the portions of the first firmware code copy, thereafter writing the given portion of the first copy of the firmware code into the random-access-memory, but

(B) if the status indicates that there are data errors in the given portion of the first firmware code copy, the boot code causes the processor to determine whether the number of bit errors in the firmware code exceed a given number, and

(i) if the number of bit errors do not exceed the given number, further causes the processor to correct the erroneous bits and write the corrected first firmware code copy into the random-access-memory, but

(ii) if the number of bit errors is equal to or exceeds the given number, further causes the processor to read at least a portion of the second firmware copy, pass the read second firmware code through the ECC calculation circuitry which calculates at least one ECC therefrom and provides a status with respect to any data errors existing in said at least a portion of the second firmware code copy to which said at least one ECC pertains, and if the status indicates that there are no data errors in said at least one portion of the second firmware code copy, thereafter writing said at least one portion of the read second copy of the firmware code into the random-access-memory.

24. (Original) The system of claim 23, wherein the firmware code is stored in the flash memory with one-bit thereof per memory cell storage element, and further wherein data are stored at at least some of addresses of the memory array other than those containing the firmware code with more than one-bit thereof per memory cell storage element.

25. (Original) A flash memory storage system, comprising:

an array of flash memory cells storing data in charge storage elements and containing at least one copy of firmware code stored therein along with a first flag indicating the presence of the firmware code and a second flag indicating that the firmware code, if present, should not be loaded,

a controller processor,

a read-only-memory containing boot code that the processor accesses and executes in response to initialization of the storage system,

a random-access-memory that is accessible by the processor to obtain instructions to be executed, and

wherein the boot code causes the processor to look for the first and second flags, and,

(A) if the first flag is present and the second flag is not present, to proceed to load the firmware code into the random-access-memory, or

(B) if both of the first and second flags are present, to provide access to the firmware code for testing without loading the firmware code into the random-access-memory, or

(C) if the first flag is not present, neither attempt to load the firmware code into the random-access-memory nor attempt to provide access to the firmware code for testing.

26. (Original) The system of claim 25, wherein the firmware code and first and second flags are stored in the flash memory with one-bit thereof per memory cell storage element, and further wherein data are stored at at least some addresses of the memory array other than those containing the firmware code and first and second flags with more than one-bit thereof per memory cell storage element.

27. (Original) A method of initiating a memory storage system having flash memory containing at least first and second copies of firmware code stored in different locations therein and user data in other locations therein, a microprocessor, a read-only-memory (ROM) containing microprocessor accessible boot code and a random-access-memory (RAM) for storing microprocessor accessible firmware code, the method comprising:

storing one bit of said at least first and second copies of the firmware code in individual storage elements of memory cells within said different locations of the flash memory and more than one bit of said user data in individual storage elements of memory cells within said other locations of the flash memory,

executing the boot code to transfer the first copy of the firmware from the flash memory to the RAM,

identifying any bit errors in the first copy of the firmware code as it is being transferred,

either correcting specific ones of the identified bit errors in the transferred first firmware code copy or reading at least a portion of the second copy of the firmware code to replace at least a portion of the first firmware code copy containing the identified bit errors, and

executing an error free copy of the firmware code from the RAM.